NASA Technical Memorandum 101633

THE POTENTIAL APPLICATION OF THE BLACKBOARD MODEL OF PROBLEM SOLVING TO MULTIDISCIPLINARY DESIGN

James L. Rogers

(NASA-TM-101633) THE POTENTIAL APPLICATION OF THE BLACKBOARD MODEL OF PROBLEM SOLVING TO MULTIDISCIPLINARY DESIGN (NASA. Langley Research Center) 12 p CSCL 09B

N89-28314

Unclas
G3/61 0224591

July 1989

National Aeronautics and
Space Administration

Langley Research Center
Hampton, Virginia 23665-5225

THE POTENTIAL APPLICATION OF THE BLACKBOARD MODEL OF PROBLEM SOLVING TO MULTIDISCIPLINARY DESIGN

J. L. Rogers*
NASA Langley Research Center
Hampton, VA

Abstract

This paper discusses the potential application of the blackboard model of problem solving to multidisciplinary design. Multidisciplinary design problems are complex, ill-structured, and lack a predetermined decision path from the initial starting point to the final solution. The final solution is achieved using data from different engineering disciplines. Ideally, for the final solution to be the optimum solution, there must be a significant amount of communication among the different disciplines plus intradisciplinary and interdisciplinary optimization. In reality, this is not what happens in today's sequential approach to multidisciplinary design. Therefore it is highly unlikely that the final solution is the true optimum solution from an interdisciplinary optimization standpoint. A multilevel decomposition approach has been suggested as a technique to overcome the problems associated with the sequential approach, but no tool currently exists with which to fully implement this technique. A system based on the blackboard model of problem solving appears to be an ideal tool for implementing this technique because it offers an incremental problem solving approach that requires no a priori determined reasoning path. Thus it has the potential of finding a more optimum solution for the multidisciplinary design problems found in today's aerospace industries.

I. Problem Background

During the design process, the design problem can be decomposed into subproblems, each pertaining to a different discipline. Ideally, for the final solution to be the optimum solution, there must be a significant amount of communication among the different disciplines plus a significant amount of both intradisciplinary and interdisciplinary optimization.

In reality, the optimum solution is seldom found in today's multidisciplinary design process because the majority of the design decisions are made in a sequence such as the one for aircraft design shown in figure 1.[1] The design sequence typically follows these steps. First, the aerodynamic engineers take the initial model, analyze it, perform some optimization, and pass a model that has been aerodynamically constrained on to the structural engineers. The structural engineers take this constrained model, analyze it, perform some optimization, and pass a further constrained model on to the next group of engineers for analysis and optimization in their particular discipline. This sequential approach to multidisciplinary design implies that iterations are to be performed until an optimum solution is obtained. However, because of budget and time constraints, very few (if any) iterations with interdisciplinary optimization are performed. Since there is little or no interdisciplinary optimization feedback, engineers seldom get a chance to analyze and optimize their discipline after changes have been made by other disciplines downstream in the design process. Thus, applying the disciplines in some sequential order and freezing their respective variables without any interdisciplinary optimization does not necessarily yield the globally optimum solution.

The globally optimum solution is not always obtained by the sequential approach to multidisciplinary design because of the following paradox posed by the approach. Paradox: as more and more time is spent on the design, the knowledge increases about the object being designed, but the freedom about which design decisions can be made decreases (see Fig. 2).[1] This is a result of constraints being imposed by the different disciplines in the design process. Thus a large percentage of the crucial design decisions that are made during the early phases of the design process limit the design freedom during the later phases.

To overcome the paradox associated with the sequential approach, Sobieski developed a multilevel decomposition approach.[1] By nature, multidisciplinary design problems are complex, ill-defined, ill-structured and lack a predetermined decision path from the initial starting point to the final solution. In this approach, a large problem is broken down into a hierarchy of smaller, self-contained subproblems which

* Computer Scientist

are simple, well-defined, and well-structured (see Fig. 3). The links among these subproblems is preserved. The linking information is small relative to the vast amount of information that must be processed internally for each subproblem. The solution efforts are guided and controlled by a hierarchy of goals and subgoals to yield a more optimum solution.

The multilevel decomposition approach proved to be only feasible when a problem could be decomposed into a strictly top-down hierarchy. Some design problems do not fall into this category because there may be a link between two or more subproblems at the same level. To address these other types of design problems, Sobieski introduced "a new method based on sensitivity analysis of a complex, coupled system which yields derivatives of the system behavior with respect to design variables fully accounting for the interactions among the parts of the systems and the disciplines that govern its design."[2] This approach is referred to as non-hierarchic decomposition.[3]

As the aerospace industry moves into the 1990's, new design tools are needed to implement both the hierarchic decomposition approach and the non-hierarchic approach coupled with sensitivity analysis to allow the knowledge about the object being designed to significantly increase without a significant reduction in design freedom. Since both approaches break a problem down into smaller subproblems, the key to this solution is to be able to simultaneously consider contributions from all the subproblems and handle the trade-offs up front. Another important research issue remaining for implementing this approach is how to incorporate human judgment, control, and creativity into the system. The blackboard model proposed in this paper appears to be a potential tool for combining the human elements with either multilevel decomposition approach to solve multidisciplinary design problems.

II. The Blackboard Model

Newell said the following about blackboards: "Metaphorically we can think of a set of workers, all looking at the same blackboard: each is able to read everything that is on it, and to judge when he has something worthwhile to add to it. This conception is just that of Selfridges' Pandemonium: a set of demons, each looking at the total situation and shrieking in proportion to what they see fits their natures..."[4]

Blackboard models have been around for several years. Nii has written an excellent overview of the evolution of blackboard models and the descriptions of some of the key applications of blackboard systems.[5,6] This section provides a brief introduction to the blackboard approach to problem solving, the three major components of the blackboard model (the knowledge sources, the blackboard data structure, and the control mechanism), and the steps in the problem solving sequence.[5,6,7] Examples of the components with respect to design are given in Section IV.

The Blackboard Approach to Problem Solving

The blackboard approach to problem solving is composed of several steps. First, the problem is divided into loosely linked subproblems where these subproblems correspond to areas of specialization. Second, the solution space is divided into different levels of abstraction of the problem and maintained on a global data base called the blackboard. Finally, the knowledge and procedures required to solve the problem are divided into specialized knowledge sources to solve the subproblems.

Nii explains the blackboard approach to problem solving in terms of putting together a jigsaw puzzle.[5] For this hypothetical example, picture a room with a large blackboard and several people each holding some pieces of a jigsaw puzzle. A few of the people volunteer to put their most promising pieces on the blackboard. Each person then looks at their pieces of the puzzle to see if they fit one of the pieces on the blackboard. Those that have pieces that fit in the current state of the puzzle add them to the blackboard. Now the group examines the current state of the puzzle and their puzzle pieces and add the pieces that fit the current state of the puzzle to the blackboard. This continues until the puzzle is complete. The puzzle can be completed without any communication among the people in the room. Each person is "self-activating" and adds pieces to the blackboard whenever there is an opportunity to do so. There is no a priori order for placing pieces on the blackboard because the order is determined by the state of the solution. Thus the solution to the puzzle is completed incrementally and opportunistically instead of systematically.

To add some control to this analogy, an aisle is added to the room so that only one person is allowed to go to the blackboard at a time. Everyone examines the current state of the puzzle and determines whether or not they can make a contribution to the solution. A monitor is needed to determine who gets to go to the blackboard. Everyone who can make a contribution to the solution raises a hand. The monitor uses some criteria for making a selection based on some strategy selected for completing the puzzle. After that piece is added, others who can possibly make a contribution raise their hands. This process continues until the problem is solved. (This serial form of the blackboard model works well for uniprocessor computers, but much research remains to be done on concurrent problem-solving methods. Nii gives a more elaborate example of the blackboard method of problem solving to explain the organization of the blackboard and the partitioning of the knowledge into knowledge sources.[5] This level of detail is beyond the scope of this paper.)

ORIGINAL PAGE IS
OF POOR QUALITY

The Knowledge Sources

The knowledge about the problem to be solved is divided into knowledge sources which are kept separate and independent. Each knowledge source can viewed as one large rule with two parts, a condition and an action. If the condition is met, then the action is executed. Therefore, knowledge sources are event-triggered because only knowledge sources with conditions evaluated to true can have their actions executed. The action part of the knowledge source may be composed of rules and/or procedures which can execute application programs. The action part of the knowledge source retrieves the appropriate data defining the current state of the problem from the blackboard and applies it to provide new information to the blackboard.

Output from the knowledge sources updates the current state of the problem solution on the blackboard and should eventually lead to a solution of the problem. Knowledge sources do not directly communicate with one another. All communication among knowledge sources is done through the blackboard. Only the knowledge sources are allowed to modify the entries on the blackboard. By communicating only through the blackboard, the knowledge sources can only indirectly influence one another when the output from an action of one knowledge source produces an entry on the blackboard which satisfies or partially satisfies the condition of another knowledge source.

The Blackboard Data Structure

The purpose of the blackboard is to provide a means of storing data that is common to more than one knowledge source, simplify communication and coordination among the knowledge sources, and insure that the solutions generated during the design process remain consistent. The blackboard acts as an intermediary for the communications and interactions among the knowledge sources by storing entries generated and needed by the knowledge sources during the problem solving process. One way of storing these entries on the blackboard is in the form of *objects*. Stefik and Bobrow define objects as entities that combine the properties of procedures[8] and data since they perform computations and save local state. Each object can store and process information, create new information, and communicate with other objects through *messages*. Messages sent between objects result in an action being taken. An object receives a message which specifies some type of operation to be performed. The object responds to the message by using its own procedures to perform the operation. In other words, message sending is similar to an indirect procedure call where the message tells the object what needs to be done without saying how to do it.

The solutions and partial solutions of the problem being solved can be organized into horizontal and/or vertical partitions. The horizontal partition of the solution space represents distinct partial solutions (or alternative solutions) to the problem. These partial solutions may overlap. They reflect different groupings of the partial solutions, such as temporal, spatial, or conceptual.

The vertical partition of the solution space (a hierarchy) distinguishes objects at different levels of abstraction and groups them as *classes* which are descriptions of one or more similar objects. The class objects contain information such as super class, class variables, instance variables. The *super class* indicates which class or classes a particular class is below in the hierarchy. A class *inherits* variables from its super classes which reduces redundancy. An *instance* of a class is an object described by a particular class. Variables in objects are used for storing the state of the solution and can be given a default value. For example, **Car** may be a class, within the super class **Vehicles**, with an instance **Model-T**. *Class variables* contain information shared by all instances of a class. *Instance variables* (sometimes called *slots*) are variables local to a particular instance. In the car example, a class variable might be **number-of-wheels**, while an instance variable might be **CarOwner**. All instances of the class car will inherit the number of wheels from the class variable while having an individual owner. *Links* are used to denote the relationships among the objects. A Model-T **is-a** car and car is **a-kind-of** vehicle are examples of links. By using the links, objects can be collected into a group of interconnected objects called a *composite object*. A car can be thought of as a composite object because it is composed of a body, power system, and electrical system.[8] The body, in turn, is composed of a hood, doors, chassis, etc. The door is composed of a window, lock, etc. Thus the lock is **a-part-of** the door which is **a-part-of** the body which is **a-part-of** the car. In addition, objects can be parts of more than one object such as an object **son** being a part of both the object **mother** and the object **father**. A composite object can then be formed into a *perspective*. Perspectives are defined as different views of the same conceptual entity. For example, depending on the perspective the same woman may be viewed as a mother, a daughter, a sister, an employee, or a supervisor.

Methods are other pieces of information are stored in classes and can be inherited from super classes. A class specifies the behavior of its instances in terms of their response to a message which is composed of a selector and arguments. A method is chosen by matching the selector with the list of methods. The methods declarations names procedures (functions) for implementing the methods. As an example, suppose the class car has a method called **Display**. **Display.Car** is a function that implements the method **Display** for instance of the class **Car** by displaying the instance of the car on the screen. The methods used are dependent upon the perspective a knowledge source has about a particular object. Entries at lower levels of abstraction may support or elaborate

on entries at higher levels. The properties of the objects at one level of abstraction serve as input to a set of the knowledge sources. The actions of these knowledge sources place new information (change the properties) of the objects at the same or different levels of abstraction. Thus the knowledge sources transform entries from one level of abstraction to another.

The Control Mechanism

Different knowledge sources may have their conditions satisfied simultaneously during the problem solving process. If this happens, a control mechanism is required to determine which knowledge source should execute next. The control mechanism consists of a set of control modules which monitor changes in the blackboard. It applies its own knowledge base about the problem being solved and the repertoire of available knowledge sources to determine the best step to take next.

This approach can apply different solution strategies to build the solution one step at a time. One strategy might be to proceed bottom-up through the different levels of abstraction. This is done by first applying only those knowledge sources that generate entries at the lowest level of abstraction. Once these are exhausted, only the knowledge sources generating entries at the next higher level are applied, and so on until a solution is obtained. Other strategies include top-down, a combination of top-down and bottom-up, or an arbitrary application of knowledge sources.

The strategy for arbitrary application of knowledge sources appears to be very flexible. At each stage of the problem solving process, any type of reasoning step might be chosen. The step might be data driven (bottom-up), goal driven (top-down), or model driven (combination). It might be determined by focusing attention on a particular aspect of the solution. The focus of attention may be which knowledge source to activate next, which blackboard objects appear promising to pursue, or some combination (which knowledge sources to apply to which objects). As a result, the sequence of applying knowledge sources which implies the sequence of steps to solve the problem is dynamic and opportunistic rather than fixed and predetermined.

The Sequence of Problem Solving Steps

The steps taken by the blackboard system to solve a problem occur in the following iterative sequence:

(1) The action part of a knowledge source makes a change(s) to an object(s) on the blackboard.

(2) The condition parts of the knowledge sources are examined to determine which knowledge sources can make a contribution to the solution of the problem.

(3) The control mechanism selects the focus of attention.

(4) If the focus of attention is a knowledge source, then a blackboard object(s) is chosen on which to process the knowledge source. This is called knowledge scheduling. If the focus of attention is a blackboard object, then a knowledge source is chosen to process the contents of that object. This is called event scheduling. If both a knowledge source and a blackboard object are chosen as the focus of attention, then the knowledge source processes the information in that object.

(5) One of the knowledge sources determines whether or not the problem solving process has been completed. If it has then the process terminates, otherwise the process is repeated beginning at step one.

III. A Candidate for the Blackboard Model?

Much work has been done in applying Artificial Intelligence tools and techniques to problems in different engineering disciplines.[9,10] Even the application of the blackboard approach to engineering problems is not a new concept. For example, Sriram has applied it to structural design and structural design integrated with construction.[11,12,13] But, the question remains as to whether or not the blackboard model can be coupled with Sobieski's multilevel decomposition approaches to yield an appropriate tool so that when it is applied to a multidisciplinary design problem such as the one found in the aerospace industry it will result in a more optimum design in less time and at less cost.

To determine whether or not a problem is a candidate for the blackboard approach requires an examination of the characteristics of the problem.[5] Is the problem complex and ill-structured? Simon defines a complex problem as one made up of a large number of parts that interact in a nonsimple way.[14] In such problems, the whole is greater than the sum of the parts where given the properties of the parts and the laws of their interactions, it is not a trivial matter to infer the properties of the whole. In addition, he defines ill-structured problems as those characterized by poorly defined goals and an absence of a predetermined decision path from the initial state to the goal state, and often a lack of well-defined criteria for determining whether or not a solution is acceptable. Other characteristics include a large solution space, the need for independent (or semi-independent) pieces of knowledge to cooperate in forming a solution, the integration of diverse information from a variety of input, and the need of an evolutionary solution requiring no a priori determined reasoning path causing the selection of what to do next to be made while the problem is being solved.

Since the above are all characteristics of the multilevel approach to solving multidisciplinary design problems, then

ORIGINAL PAGE IS
OF POOR QUALITY

the blackboard's incremental and opportunistic problem-solving approach appears to be an ideal tool for implementation. The blackboard system proposed in the next section as an implementation tool does not yet exist. It is meant to serve as a guide for showing the potential advantages that might be obtained from applying such a system.

IV. Implementation in the Blackboard Model

The blackboard model offers the design manager a method for keeping track of the design options and incomplete design ideas. This can be especially useful as the design progresses through a series of transitions from one design state to another. The states are adjusted based on changes that have just occurred. Since there are many different paths that may be taken, it is necessary to keep a history of the design decisions in case the design manager needs to backtrack because of some problem encountered on the current design path.[15] Keeping track of the knowledge of incomplete design ideas is important because design is a dynamic and ever changing process where an alternative that is valid at one state may be invalid at another and vice-versa. The blackboard model can also keep track of the reasoning behind the design decisions and of the general evolution of the design process. This reasoning can be captured as design experience and applied to improve future designs.[16] This could result in significant savings since estimates are that 90% or more of the things that are designed are actually redesigns of something that already exists.[17]

To implement the multilevel decomposition approach in a blackboard model, the problem must first be decomposed into design subproblems. This can be done with tools such as the one developed by Rogers.[18] This is important because the design manager must know how to group the modules into subproblems and how to assign the subproblems to design teams so that changes in one subproblem will have predictable effects on other subproblems. The decomposition of a generic design problem is shown in figures 4 and 5. Figure 4 displays the group of modules before decomposition. The modules are represented by boxes along the diagonal. A horizontal line from the box indicates an output from the module, while a vertical line indicates an input to the module. A vertical line entering the module from below implies a feedback link. The small circles indicate a link between two modules where the output from one module is an input to another module. The links among the modules consist of design variables, behavior variables, and constraints. At this point, there is very little information about the links among the modules that might be of benefit to a design manager. Figure 5, on the other hand, displays the same group of modules after they have been reordered and decomposed into subproblems by Rogers' decomposition tool. The purpose of this tool is to limit the number of feedback links thereby decreasing the cost of obtaining a solution by partitioning the modules of a problem into circuits which represent subproblems. All feedback links are contained within the circuits indicating that an iteration is required. Circuits are connected to each other only by feedforward links, therefore there is no iteration among circuits and they can be ordered into a multilevel hierarchy (fig. 6) which should result in some information about how the knowledge sources will interact with the levels of abstraction on the blackboard.

The Knowledge Sources

For the blackboard model, each one of these decomposed subproblems (circuits) and its associated modules are assigned as a disciplinary knowledge source. The knowledge sources can be on the same or different computers. They are triggered when the conditional part of their rule is satisfied. The conditional part knows what objects on the blackboard a specific knowledge source is interested in, when those objects change, and determines if it can contribute some data to update those objects. The action part of the knowledge source may be a knowledge base, or an application program (analysis and intradisciplinary optimization), a database system, or some combination. The action part should also have some sort of user and graphics interface. The action part of these knowledge sources provide data from the different disciplines to the blackboard. They know which objects on the blackboard they can update and how to update them. Each disciplinary knowledge source views the global model from its own perspective and may have rules and data about that perspective local to the knowledge source. An example of a structures knowledge source is show in figure 7.

In addition to the disciplinary knowledge sources, other type of knowledge sources could prove to be useful. These knowledge sources include knowledge about interdisciplinary optimization, strategy, interaction constraints, sensitivity analysis, and resources. The interdisciplinary-optimization knowledge source, when triggered into action, would use design variables and constraints from all disciplines to provide an optimum design from a global perspective. This will probably cause changes in certain objects monitored by disciplinary knowledge sources and trigger them into action. The strategy knowledge source keeps track of the current state of the solution and uses its knowledge base to move to the next state in the solution process. The interaction-constraints knowledge source increases communication, coordination, and cooperation among the various disciplines by resolving the conflicts that arise among their constraint interfaces. The sensitivity-analysis knowledge source aids in deciding how to modify a design by determining which design variables are the most influential and whether their influence is positive or negative. And finally, the resources knowledge source contains the corporate knowledge about old designs and design decisions that can be used to guide and control the solution process.

The Blackboard and Control Mechanism

The blackboard may have more than one partition, but for this application it is assumed that only one partition, the solution partition, is to be created. The multilevel decomposition scheme decomposes the problem into a hierarchy of subproblems which can be mapped into levels of abstraction on the blackboard. A knowledge source may receive a signal for its conditional part from an object at one level and issue a change to an object at another level.

The boxes in figure 3 represent one way to decompose a problem into composite objects. As mentioned before, the objects contain information such as super class, class variables, instance variables, and methods. In the example, *wing* is a super class of *ailerons*. The two classes are linked together by ailerons are **a-part-of** the wing. Class variables in objects are used for storing the state of the solution and can be given a default value which would be useful for providing an initial design. Classes are made of specific instances. For the class wing, an instance is *left-wing*. Instance variables contain information about a specific instance of a class, such as design variables affecting that particular instance. A class specifies the behavior of its instances in terms of their response to messages. An example of a method that might be in the class wing is *Optimize Structure.Wing* which would invoke a knowledge source to optimize the wing from a structures perspective and return a change in the design variables which would be inherited by each instance of the wing class. The methods used are dependent upon the perspective a knowledge source has about a particular object. For example, the structures discipline views a wing from a different perspective than the controls discipline.

Control of the whole process is handled by the inferencing mechanism of choice. The inferencing mechanism may be developed in-house or purchased from a vendor. This mechanism keeps track of all entries on the blackboard and determines a priority for executing the knowledge sources that have their condition parts satisfied at any one point in time. From these facts, it uses its knowledge base to determine the focus of attention. It then determines which blackboard object and knowledge source will be used to move to the next solution state based on the computed priorities. Figure 8 displays the proposed blackboard implementation.

V. Summary

New software tools need to be developed for multidisciplinary design to improve productivity by achieving a more optimal design at less cost in less time. This paper presents the problems of today's sequential approach to multidisciplinary design and offers the blackboard model as a potential tool for implementing the multilevel decomposition approach to overcome those problems. The blackboard model serves as a global database for the solution with each discipline acting as a knowledge source for updating the solution. This approach applies opportunistic problem solving which requires no a priori determined reasoning path. Selection of which knowledge source to apply to move to a new problem state is achieved by examining the current state of the problem and choosing the best step from several possibilities. By using this approach it is possible that engineers can improve the coordination, communication, and cooperation in the conceptual design process allowing them to examine more alternatives, capture the design decisions for future reference, and achieve a more optimal design from an interdisciplinary viewpoint.

VI. Acknowledgments

The author wishes to thank Jean-Francois M. Barthelemy for his suggestions on the engineering aspects of this paper, and Kathy H. Abbott for her suggestions on the blackboard aspects.

VII. References

1. Sobieszczanski-Sobieski, J.; Barthelemy, J-F. M.; and Giles, G. L.: Aerospace Engineering Design by Systematic Decomposition and Multilevel Optimization. NASA TM 85823, June 1984.

2. Sobieszczanski-Sobieski, J.: "Sensitivity Analysis and Multidisciplinary Optimization for Aircraft Design: Recent Advances and Results." NASA TM 100630, July 1988.

3. Sobieszczanski-Sobieski, J.: "Optimization by Decomposition: A Step from Hierarchic to Non-hierarchic Systems." Proceedings of the Symposium on Recent Advances in Multidisciplinary Analysis and Optimization, Hampton, VA, September 28-30,1988, NASA CP-3031, Part 1, pp. 51-78.

4. Newell, A.: "Some Problems of Basic Organization in Problem Solving Programs." Proceedings of the Conference on Self-Organizing Systems, Washington, DC, 1962, pp. 393-423.

5. Nii, H. P.: "Blackboard Systems: The Blackboard Model of Problem Solving and the Evolution of Blackboard Architectures - Part 1." AI Magazine, Summer 1986, pp. 38-53.

6. Nii, H. P.: "Blackboard Systems, Blackboard Application Systems, Blackboard Systems from a Knowledge Engineering Perspective." AI Magazine, August 1986, pp. 82-106.

7. Hayes-Roth, B.: "A Blackboard Architecture for Control." Artificial Intelligence, Vol. 26, 1985, pp.251-321.

8. Stefik, M.; and Bobrow, D.: "Object-Oriented Programming: Themes and Variations." The AI Magazine, Vol. Winter, PP. 40-62, 1985.

9. Sriram, D.: A Bibliography of Knowledge-based Expert Systems in Engineering. SIGART Newsletter, No. 89, July 1984, pp. 131-136.

10. Sriram, D.; and Joobbani, R.: Special Issues, AI in Engineering. SIGART Newsletter, No. 92, April 1985, pp. 38-144.

11. Sriram, D.: "DESTINY: A Model for Integrated Structural Design." International Journal for AI in Engineering, October 1986.

12. Sriram, D.: Knowledge-Based Approaches for Structural Design. CM Publications, UK, 1987.

13. Sriram, D.; Logcher, R.; and Groleau, N.: "A Framework for a Computer Integrated Design and Construction System." Proceedings of the International Conference on Computational Engineering Science, April 10-14, 1988, Atlanta, GA, Vol. 2, pp. 62.i.1-62.i-4.

14. Simon, H. A.: The Sciences of the Artificial. MIT Press, 1969.

15. Kalay, Y. E.; Swerdloff, L. M.; and Harfmann, A. C.: "A Knowledge-based Computable Model of Design." Proceedings of the IFIP WG 5.2 Working Conference on Expert Systems in Computer Aided Design, Sydney Australia, February 1987, pp. 203-227.

16. Maher, M. L.; and Zhao, F.: "Using Experience to Plan the Synthesis of New Designs." Proceedings of the IFIP WG 5.2 Working Conference on Expert Systems in Computer Aided Design, Sydney Australia, February 1987, pp. 349-373.

17. Dietterich, T. G.; and Ullman, D. G.: "FORLOG: A Logic-based Architecture for Design." Proceedings of the IFIP WG 5.2 Working Conference on Expert Systems in Computer Aided Design, Sydney Australia, February 1987, pp. 1-24.

18. Rogers, J. L.: " A Knowledge-based Tool for Multilevel Decomposition of a Complex Design Problem." NASA TP 2903, January 1989.

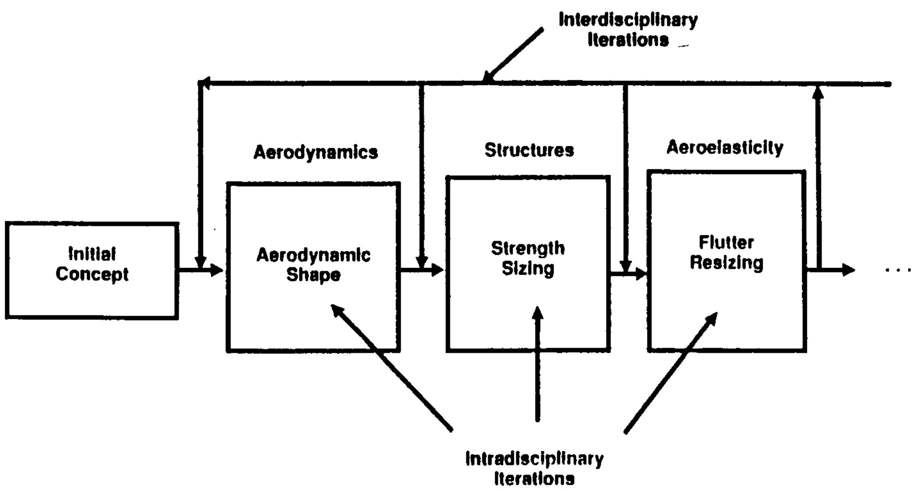

Fig. 1 Sequential decision making in the design process freezes out subsequent design freedom.

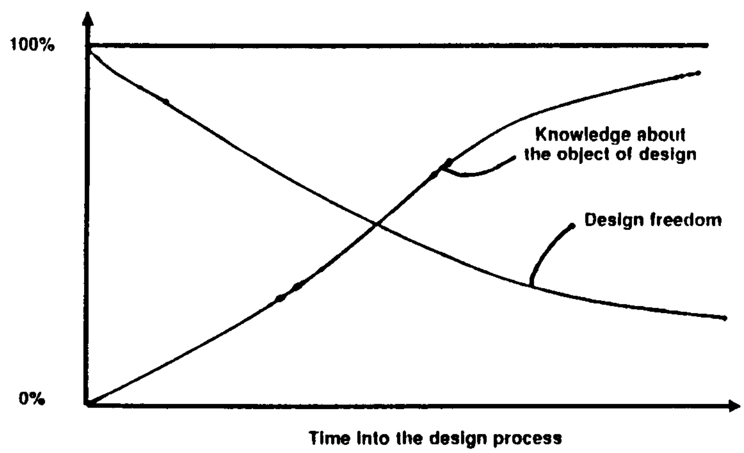

Fig. 2 Paradox of the conventional sequential decision making in design.

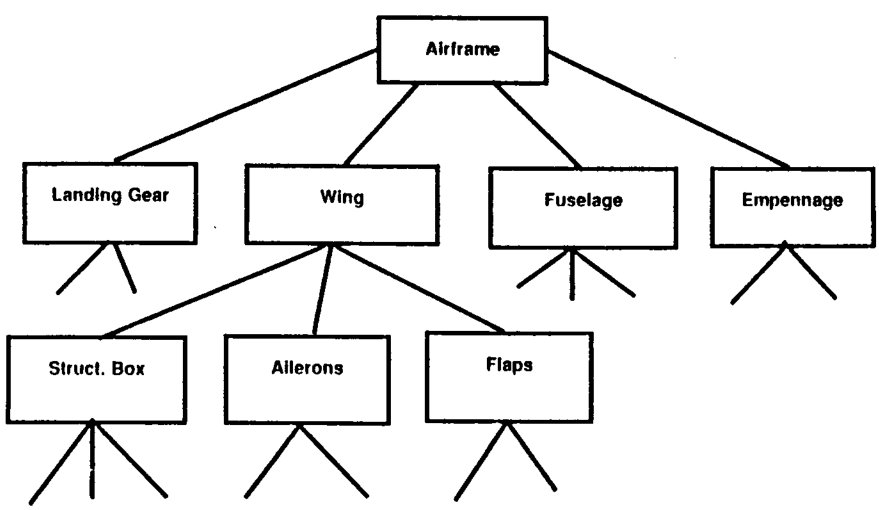

Fig. 3 Typical decomposition of a design problem.

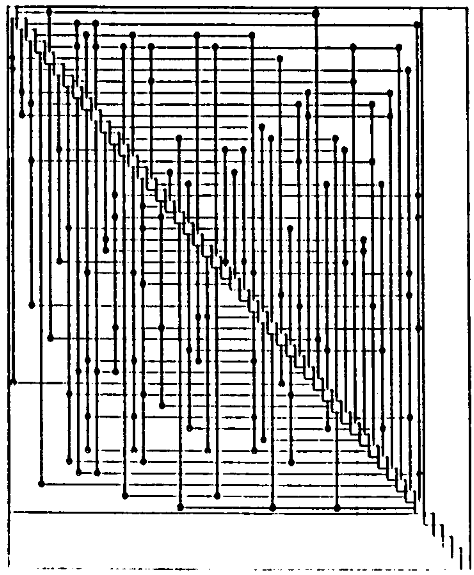

Fig. 4 Nondecomposed problem.

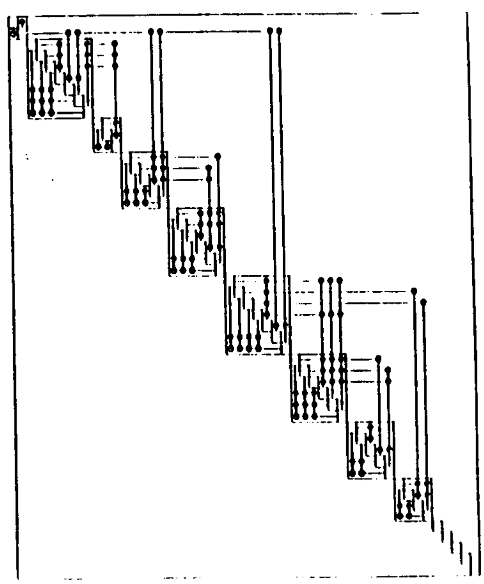

Fig. 5 Modules and circuits.

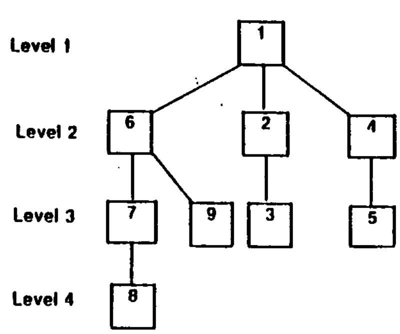

Fig. 6 Decomposed problem.

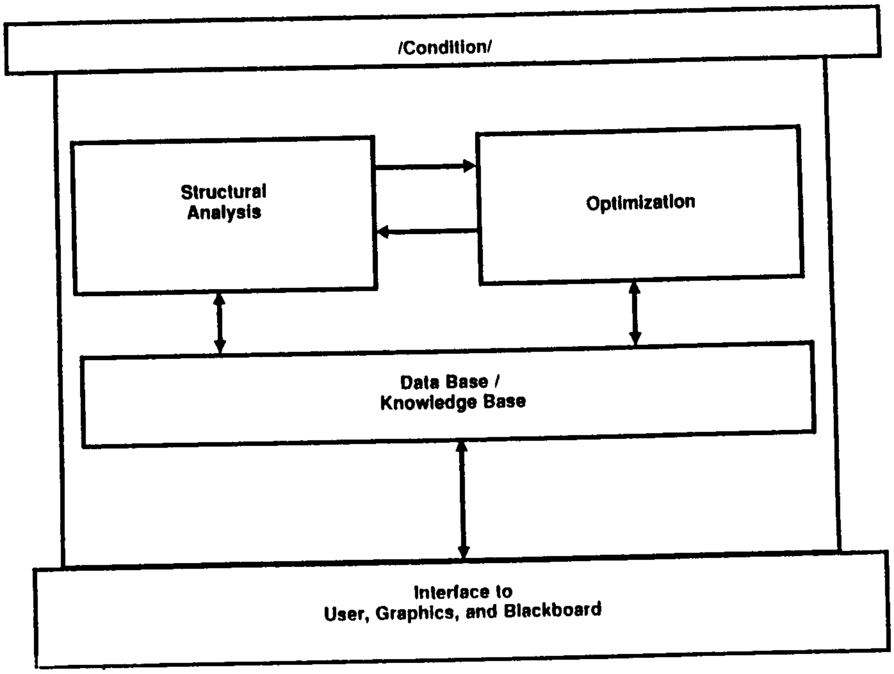

Fig. 7 Sample structures knowledge source.

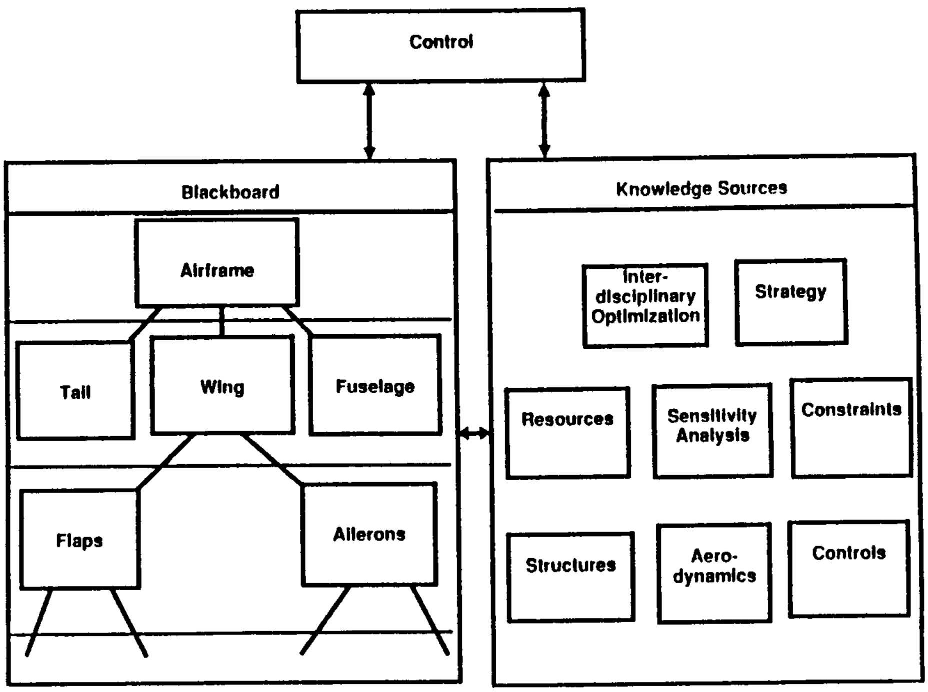

Fig. 8 Blackboard implementation of the multilevel decomposition approach.

NASA
National Aeronautics and Space Administration

Report Documentation Page

1. Report No.	2. Government Accession No.	3. Recipient's Catalog No.
NASA TM-101633		

4. Title and Subtitle
The Potential Application of the Blackboard Model of Problem Solving to Multidisciplinary Design

5. Report Date
July 1989

6. Performing Organization Code

7. Author(s)
James L. Rogers

8. Performing Organization Report No.

10. Work Unit No.
505-63-01-07

9. Performing Organization Name and Address
NASA Langley Research Center
Hampton, VA 23665-5225

11. Contract or Grant No.

13. Type of Report and Period Covered
Technical Memorandum

12. Sponsoring Agency Name and Address
National Aeronautics and Space Administration
Washington, DC 20546-0001

14. Sponsoring Agency Code

15. Supplementary Notes
To be presented at the AIAA Computers in Aerospace VII Conference, Monterey, CA, October 3-5, 1989. (AIAA Paper No. 89-3071)

16. Abstract
This paper discusses the potential application of the blackborad model of problem solving to multidisciplinary design. Multidisciplinary design problems are complex, ill structured, and lack a predetermined decision path from the initial starting point to the final solution. The final solution is achieved using data from different engineering disciplines. Ideally, for the final solution to be the optimum solution, there must be a significant amount of communication among the different disciplines plus intradisciplinary and interdisciplinary optimization. In reality, this is not what happens in today's sequential approach to multidisciplinary design. Therefore it is highly unlikely that the final solution is the true optimum solution from an interdisciplinary optimization standpoint. A multilevel decomposition approach has been suggested as a technique to overcome the problems associated with the sequential approach, but no tool currently exists with which to fully implement this technique. A system based on the blackboard model of problem solving appears to be an ideal tool for implementing this technique because it offers an incremental problem solving approach that requires no a priori determined reasoning path. Thus it has the potential of finding a more optimum solution for the multidisciplinary design problems found in today's aerospace industries.

17. Key Words (Suggested by Author(s))
multidisciplinary design
Blackboard Model
Decomposition

18. Distribution Statement
Unclassified - Unlimited
Subject Category 61

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of pages	22. Price
Unclassified	Unclassified	11	A03

NASA FORM 1626 OCT 86